



[> home](#) [> about](#) [> feedback](#) [> login](#)  
US Patent & Trademark Office

## Search Results

### Nothing Found

Your search for the *Phrase* `stewart <near/1> sabadell` did not return any results.  
To search for *terms* separate them with AND or OR.  
Click on the suggested options:

`stewart AND <near/1> AND sabadell`

`stewart OR OR sabadell`

To search for names try using only the last or first name.

You can try to rerun it within the Portal.

You may revise it and try your search again below or click advanced search for more options.

<code>stewart &lt;near/1&gt; sabadell</code> <input type="checkbox"/>	<input type="button" value="SEARCH"/> <a href="#">[Advanced Search]</a> <a href="#">[Search Help/Tips]</a>
--	--

 Complete Search Help and Tips

### The following characters have specialized meaning:

Special Characters	Description
, ( ) [	These characters end a text token.
= > < !	These characters end a text token because they signify the start of a field operator. (! is special: != ends a token.)
` @ \Q < { [ !	These characters signify the start of a delimited token. These are terminated by the end character associated with the start character.



[> home](#) [> about](#) [> feedback](#) [> login](#)  
US Patent & Trademark Office

## Search Results

### Nothing Found

Your search for the *Phrase* nikolai <near/1> sander did not return any results.  
To search for *terms* separate them with AND or OR.  
Click on the suggested options:

nikolai AND <near/1> AND sander

nikolai OR OR sander

To search for names try using only the last or first name.

You can try to rerun it within the Portal.

You may revise it and try your search again below or click advanced search for more options.

nikolai <near/1> sander	SEARCH	[Advanced Search]	[Search]
-------------------------	--------	-------------------	----------

### Help/Tips]

Complete Search Help and Tips

### The following characters have specialized meaning:

Special Characters	Description
, ( ) [	These characters end a text token.
= > < !	These characters end a text token because they signify the start of a field operator. (! is special: != ends a token.)
` @ \Q < { [ !	These characters signify the start of a delimited token. These are terminated by the end character associated with the start character.



[> home](#) [> about](#) [> feedback](#) [> login](#)  
US Patent & Trademark Office

## Search Results

Search Results for: [john <near/1> hutchinson ]  
Found 5 of 105,557 searched. → Rerun within the Portal

Search within Results

---

**GO** [> Advanced Search](#) [> Search Help/Tips](#)

---

**Sort by:** [Title](#) [Publication](#) [Publication Date](#) [Score](#) [Binder](#)

---

**Results 1 - 5 of 5** [short listing](#)

---

**1** Is Pascal a logical subset of Algol68 or not? Part One 100%

Roy Edwards

Proceedings of the Strathclyde ALGOL 68 conference March 1977

It is often believed that Pascal is - Algol68 in miniature - well  
&ldquo;structured&rdquo;. It will be argued in Part One of this paper  
that both beliefs are badly founded.

**2** The equivalence of models of tasking 100%

Daniel M. Berry

Proceedings of ACM conference on Proving assertions about programs  
January 1972

A technique for proving the equivalence of implementations of  
multi-tasking programming languages is developed and applied to  
proving the equivalence of the contour model and a multi-tasking  
version of the copy rule.

**3** Interfaces between protocol layers on a multiprocessor system 100%


Wilfredo A. Colon-Castro , Deborah A. Kirkman

Proceedings of the 8th Data Communications Symposium October 1983


A multi-layer, X.25 based protocol has been implemented on a  
multiprocessor system. The higher layer protocol resides in the  
system processor while the lower level protocol (X.25 levels 2 and 3)  
runs on a peripheral processor. This paper focuses on the boundary  
between the system and peripheral processors. It discusses how flow  
control is done, how control information is passed across the  
boundary, and the synchronization problems that arose. In addition,

it describes effects on performance ...

**4** Construction of fractal objects with iterated function systems 100%

 Stephen Demko , Laurie Hodges , Bruce Naylor  
ACM SIGGRAPH Computer Graphics , Proceedings of the 12th annual  
conference on Computer graphics and interactive techniques July 1985  
Volume 19 Issue 3

**5** A parallel iterative linear system solver with dynamic load 100%

 balancing  
Peter Christen  
Proceedings of the 12th international conference on Supercomputing  
July 1998

---

**Results 1 - 5 of 5      short listing**

---

The ACM Portal is published by the Association for Computing Machinery. Copyright © 2003 ACM, Inc.